

MICRO CONTROLLER BOARDS

PS-8085 Trainer Kit



**USER MANUAL
& TECHNICAL REFERENCE**

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION	5
1. INTRODUCTION	5
1.2 PS – 8085 BOARD OVERVIEW	6
1.3 PS – 8085 SPECIFICATIONS.....	7
CHAPTER – 2: SYSTEM DESCRIPTION	9
2.1 HARDWARE DESCRIPTION.....	9
2.2 CONNECTOR DETAILS.....	12
1) 20 pin expansion connectors:	12
2) 50 pin expansion connector:.....	13
The 50 Pin FRC connector is used to interconnect with the Interface cards like 8255, 8279, 8253/8251, 8259, 8257 and the pin details are given below	13
2.3 POWER SUPPLY DETAILS	17
2.4 KEYBOARD DETAILS.....	17
CHAPTER – 3 COMMANDS AND KEYS	18
3.1 RESET.....	18
3.2 H (Help Menu).....	18
CHAPTER – 4 OPERATING INSTRUCTIONS.....	19
4.1 POWER ON	19
4.2 PROGRAM ENTRY USING ASSEMBLER:	20

Join the Technical Community Today!

<http://www.pantechsolutions.net>

1) ENTERING MNEMONICS.....	20
ENTERING 'G 'EXECUTING COMMAND	23
4.3 ENTERING RESULT COMMAND:	24
4.4 DISASSEMBLER	24
4.5 M (MODIFY EXTERNAL MEMORY):.....	25
4.6 R (REGISTER DISPLAY)	26
4.7 T (TRANSFER COMMAND).....	27
4.8 N (LOCAL MODE)	28
4.9 B (BAUD RATE)	28
4.10 S (SERIAL MODE KEY)	29
1) Initially connect the 9V adaptor to J8 connector	30
REGISTERS:	35
CHAPTER 6: EXAMPLE PROGRAMS	36
6.1 8 BIT MULTIPLICATION OF TWO DATA'S.....	37
6.2 16 BIT MULTIPLICATION OF TWO DATA'S.....	40
6.3 8 BIT DIVISION OF TWO DATA'S	42
6.4 16 BIT DIVISION OF TWO DATA'S	45
6.5 2 DIGIT BCD TO BINARY (HEXA decimal) CONVERSATION	47
6.6 BINARY (Hex) TO ASCII CONVERSION.....	50
6.7 ASCENDING ORDER	54
6.8 DESCENDING ORDER	57
6.9 FIND THE BIGGEST NUMBER IN AN ARRAY	60

6.10	VERIFYING THE PORTS OF 8255	62
6.11	SQUARE WAVE GENERATION USING 8253/54	
		64

CHAPTER 1: INTRODUCTION

1. INTRODUCTION

The PS-8085 board which demonstrates the capabilities of the 40-pin 8085 (various families) Sample programs are provided to demonstrate the unique features of the supported devices.

The PS-8085 Board Kit comes with the following:

1. PS-8085 Board
2. Sample devices (INTEL 8085/NEC 8085)
3. Cross cable (RS232)
4. CD-ROM, which contains:
 - a) Sample programs
 - b) PS-8085 Board User manual
5. User and Technical reference manual
6. Keyboard (101 keys)

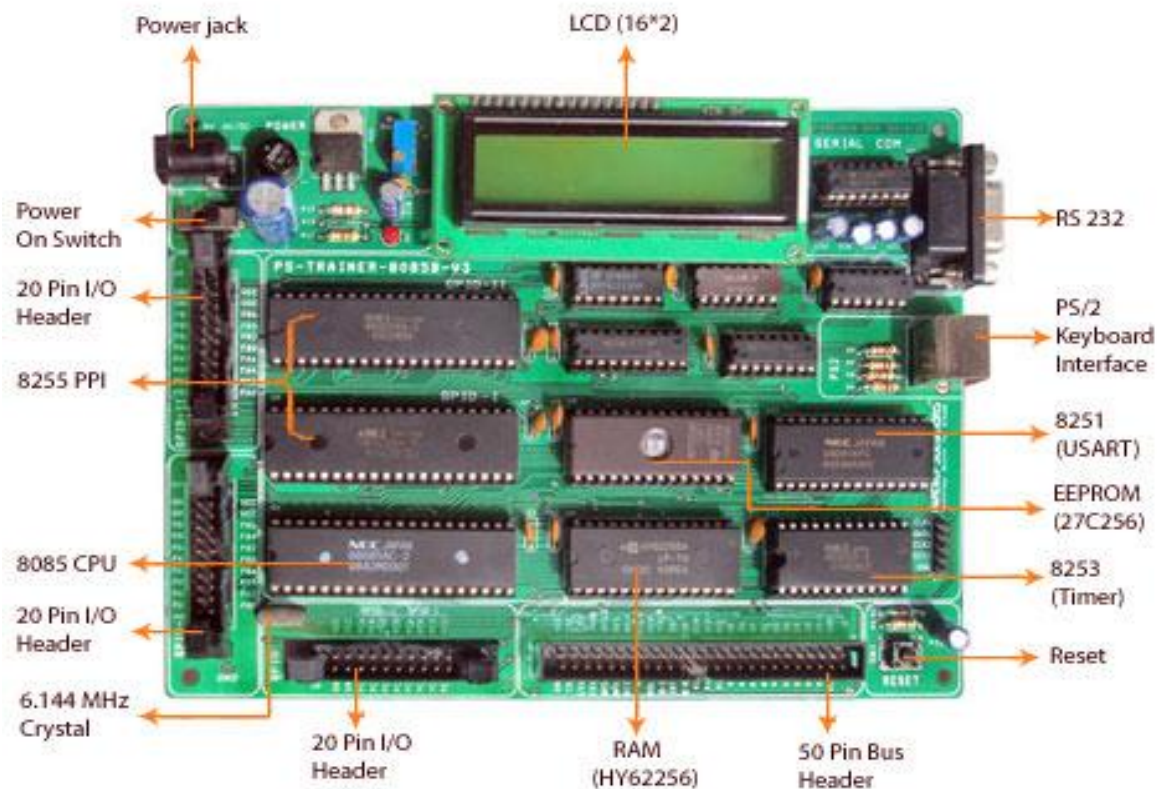
Join the Technical Community Today!
<http://www.pantechsolutions.net>

7. 20 pin FRC cable

8. 50 pin FRC cable

Note: If you are missing any part of the kit, please contact our support executive

1.2 PS – 8085 BOARD OVERVIEW



Join the Technical Community Today!
<http://www.pantechsolutions.net>

The PS-85 board is based on Intel 8085 Microprocessor, which operates at 6.144 MHz. The board can operate using the 101/104 PC keyboard supplied along with the trainer kit and 2 Line by 16-character LCD display or from the PC.

Microprocessor's Address, Data and Control bus pins are brought to the 50-pin FRC connector. PS-85 is equipped with powerful software monitor in 27C128 EPROM.

The monitor supports Video terminals RS232C interface, 101 keyboard and LCD display. The board has 32KB CMOS static RAM (type 62256),

1.3 PS – 8085 SPECIFICATIONS

- **8085** Microprocessor operating at **6.144 MHz**

- **16KB** powerful software monitors **27C128 EPROM**.
- Three **16** bit programmable timers from **8253/8254**.
- **48** Programmable I/O lines from two nos. of **8255**.
- Serial interface using **8251**.
- **50** Pin **FRC** connector for system bus expansion.
- **20** pin **FRC** connectors for user interface from **8255**.
- **9** Pin D type female connector for **RS 232C** interface and 9 Pin.
- **Six numbers** of selectable baud rates from **150 to 9600**.
- **101 PC** type keyboard for entering user address/data and for commands.
- Built in line-by-line **Assembler** and **Disassembler**.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

- User-friendly software monitor for loading and executing programs with break point facility.
- Facility to connect to **PC**.

CHAPTER – 2: SYSTEM DESCRIPTION

2.1 HARDWARE DESCRIPTION

PROCESSOR, CLOCK FREQUENCY:

8085 operates at 16.144 MHz clock.

MEMORY:

Monitor EPROM : 0000 – 3FFF (SEGMENT)

System RAM : 8000 – 8FFF (SEGMENT)

A000 – FFFF (Reserved for Monitor program)

User RAM Area : 8500 – FFFF

Join the Technical Community Today!
<http://www.pantechsolutions.net>

ALLOCATION OF EPROM:

START ADDRESS	END ADDRESS	SOCKET NO	IC USED	TOTAL CAPACITY
0000	3FFF	U3	27256	32 K BYTE

ALLOCATION OF RAM:

START ADDRESS	END ADDRESS	SOCKET NO	IC USED	TOTAL CAPACITY
0000	FFFF	U4	62256	32 K BYTE

PARALLEL INTERFACE:

8255 - Programmable peripheral interface.

SYSTEM MAPPING: I/O mapped I/O.

The following are the I/O addresses for 8255(GPIO I):

SOCKET.NO	FUNCTION	ADDRESS	CONNECTOR.NO
U8	CONTL REG	23	J1 GPIO I J5 (GPIO I&GPIOII)
	PORT A	20	
	PORT B	21	
	PORT C	22	

The following are the I/O addresses for 8255(GPIO II):

SOCKET.NO	FUNCTION	ADDRESS	CONNECTOR.NO
U9	CONTL REG	33	J4 GPIO II J5 (GPIO I&GPIOII)
	PORT A	30	
	PORT B	31	
	PORT C	33	

TIMER INTERFACE:

8253 - Programmable Interval Timer:

SYSTEM MAPPING: I/O mapped I/O.

CHANNEL 2:

Input clock : 3 MHz
Output clock : Depends on selection of baud rate.
Used for : Baud rate generation for 8521
USART.

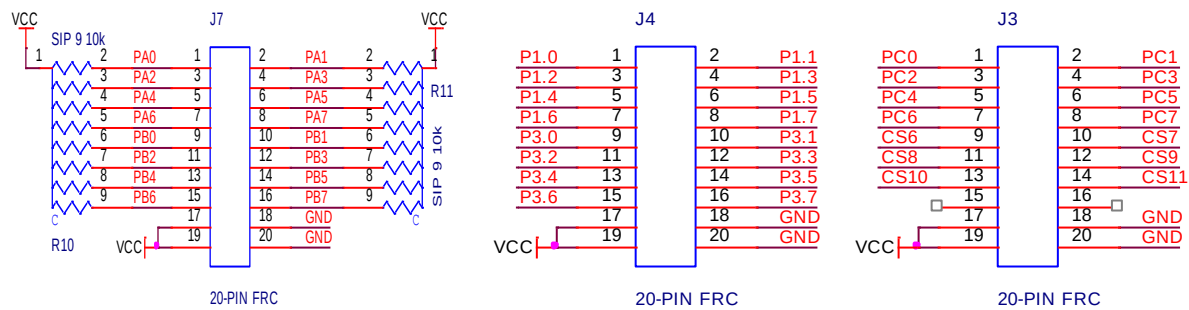
I/O ADDRESS:

SOCKET.NO	FUNCTION	ADDRESS	CONNECTOR.NO
U10	CONTRL REG	13	J7
	CHENNAL 0	10	
	CHENNAL 1	11	
	CHANNEL 2	12	

2.2 CONNECTOR DETAILS

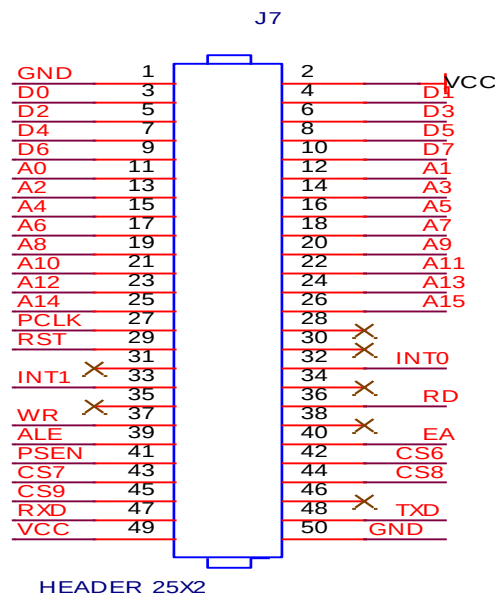
1) 20 pin expansion connectors:

A 20 Pin FRC connector is used to interconnect with the Interface cards like ADC, DAC, SWITCH/LED, RELAY buzzer Interfaces etc. Pin details are given below

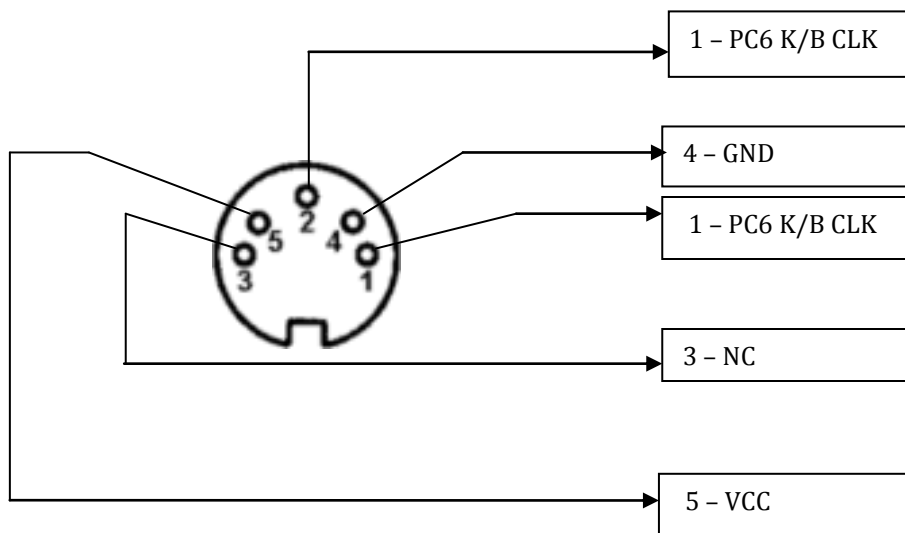


2) 50 pin expansion connector:

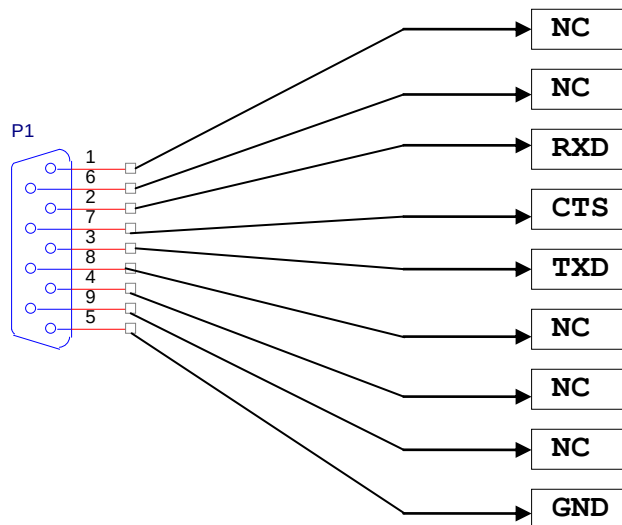
The 50 Pin FRC connector is used to interconnect with the **Interface cards** like 8255, 8279, 8253/8251, 8259, 8257 and the **pin details** are given below



3) KEYBOARD CONNECTOR:



4) 9PIN 'D' TYPE (FEMALE):



8251 - UNIVERSAL SYNCHRONOUS / ASYNCHRONOUS RECEIVER / TRANSMITTER.

BAUD CLOCK:

Baud clock for 8251 is programmable,
provided by Channel 2 of 8253

INPUT CLOCK FOR 8251:

3.072 MHz

DRIVERS USED:

MAX 232 is used for transmitting receiving of
characters.

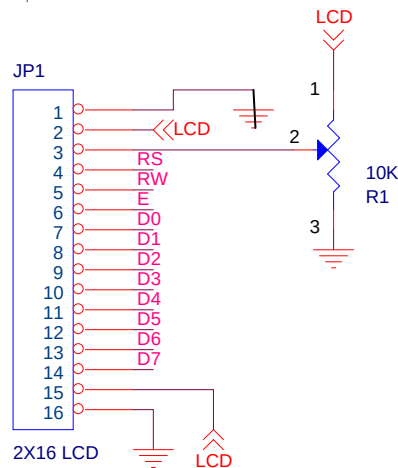
8251 UART I/O ADDRESS:

SOCKET.NO	FUNCTION	ADDRESS	CONNECTOR.NO
U11	8251 CONTL REG 8251 DATA	00 01	D2(SKT)

Join the Technical Community Today!
<http://www.pantechsolutions.net>

5) LCD INTERFACE:

DEVICE USED: 16 × 2 / 20 × 4 LCD module



SYSTEM MAPPING: I/O mapped I/O.

SOCKET.NO	FUNCTION	ADDRESS	CONNECTOR.NO
----	LCD COMMAND LCD DATA	40 41	-----

6) RESET:

This key is located in the main 8086 board. On depressing this key the program starts executing from the

beginning or reset address 0000. On power on reset it.
Display **PAN -85** in local LCD display.



2.3 POWER SUPPLY DETAILS

PS trainer kit will work at 0 – 9v (1 amp) from the PS power supply. Provision is made in PS power supply to bring out on the front panel DC regulated voltage output for interfacing with add-on cards.

+9V 1 amp

POWER SWITCH		Supply Turned OFF
		Supply Turned ON

2.4 KEYBOARD DETAILS

101 PC type keyboard is interfaced to Microcontroller through its port pins. Communication between keyboard

Join the Technical Community Today!
<http://www.pantechsolutions.net>

and Microcontroller takes place using 2 wires- one for serial clock and serial data.

CHAPTER – 3 COMMANDS AND KEYS

3.1 RESET

This key is located in the main 85L board. On depressing this key the program starts executing from the beginning or reset address 0000. On power on reset it. Display **PS - 85** in local LCD display.

3.2 H (Help Menu)

This key is used go PS-85L help menu and it will display the following commands. Depressing the *space bar key* increments to display the next command.

KEY	FUNCTIONS
A <BEG>	ASSEMBLE
B	BAUD RATE
D <BEG>	DISASSEMBLE
E <BEG><END>	EXAMINE
G <BEG>	EXECUTE
H	HELP COMMANDS
L <OFFSET>	DOWN LOAD
M <ADDR>	MODIFY
N	NORMAL MODE
Q	QUIT
R <REG>	REGISTER DISPLAY
S	SERIAL MODE
T <START><END><DS>	BLOCK TRANSFER
U <BEG><END>	UP LOAD

CHAPTER – 4 OPERATING INSTRUCTIONS

4.1 POWER ON

Connect the PS – 8051 board to the power having the following specifications.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

+9V DC 1 Amp

Switch on the power supply after ensuring the correct voltages. Following message will appear on the LCD display.

PAN 85

On power on or after reset the display shows **PAN – 85** as a sign on message. The prompt character – is displayed in the next line informing the user, that the board is ready to accept the commands.

4.2 PROGRAM ENTRY USING ASSEMBLER:

1) ENTERING MNEMONICS

Example:

Press H for help

_A8500

Enter the starting Address

Enter Key

User program from start address 8500. Display the following and waits for the user data to be typed in the second line.

Example:

8500:

MVI A,12

Enter The Mnemonics

Enter key

8502:

MVI B,12

Enter The Mnemonics

Enter key

Program end.

EXIT COMMAND: Double Enter you get the main menu

Join the Technical Community Today!
<http://www.pantechsolutions.net>

**Press H for
help
—**

2) PROGRAM ENTRY USING OPCODE:

Modify External memory

**Press H for
help
_M8500**

Enter the starting Address

Enter Key ↵

**8500:
4A —**

**8500:
4A 3E**

Enter the Opcode

Enter the Space Bar Key

8501:
56 12

Enter the Opcode

Enter the Space Bar Key

EXIT COMMAND: Double Enter you get the Main Menu

ENTERING 'G 'EXECUTING COMMAND

**Press H for
help
_G8500**

Enter starting address

Enter Key 

After executing DISPLAY

—

Executing display

To Exit Execution Mode PRESS 'RESET 'Switch

Join the Technical Community Today!
<http://www.pantechsolutions.net>

4.3 ENTERING RESULT COMMAND:

**Press H for
help**

_M

<address>

<Address>:

24 _

Enter the memory location

Enter key ↵

You get the output
8-bit data

4.4 DISASSEMBLER

Disassemble converts the hex byte stored in the memory into equivalent mnemonics. To enter into disassemble mode, type D in the command mode followed by the memory address.

EXAMPLE:

**Press H for
help**

_D8500

Enter the starting Address

Enter Key ↵

```
8500:    3E
12
MVI A,12
```

Enter the Space Bar Key

```
8500:    06
12
MVI B,12
```

Enter the Space Bar Key

4.5 M (MODIFY EXTERNAL MEMORY):

Using this command the user can display/modify any external memory address.

Modify External Memory

Join the Technical Community Today!
<http://www.pantechsolutions.net>

**Press H for
help
_M8500**

Enter the Starting Address

4.6 R (REGISTER DISPLAY)

EXAMPLE:

**Press H for
help
_R**

Enter the Starting Address

Enter Key ↵

A = FF

Enter the Space Bar Key

B = FF

Enter the Space Bar Key

Join the Technical Community Today!
<http://www.pantechsolutions.net>

4.7 T (TRANSFER COMMAND)

EXAMPLE:

T8500 8600 9000 Press Enter Key.

The above command transfer the memory content starting from source start address 8500 to destination start address 8600 till source end address 9000 is reached.

Starting address	8500
End address	8600
Destination address	9000

**Press H for
help
_T8500
8600 9000**

Enter Key 

Join the Technical Community Today!
<http://www.pantechsolutions.net>

**Transfer
Complete**

Enter Key for Exit Command

4.8 N (LOCAL MODE)

When this key is depressed on PC keyboard, the PS – 8051 Kit starts working through local 101 keyboard. Serial communication is disabled. Following message will appear in the LCD display.

**! Normal
Mode!**

4.9 B (BAUD RATE)

Press the 'B' Baud rate will be displayed on LCD

**Cur Baud =
2400
150**

Enter the Space Bar Key

Join the Technical Community Today!
<http://www.pantechsolutions.net>

**Cur Baud =
2400
9600**

Enter Key ↵ Set the 9600 Baud Rate

Baud Rates : 150, 300, 600, 1200, 2400, 4800, **9600**

4.10 S (SERIAL MODE KEY)

When this key is depressed the system starts communicating through RS232C connector. All keys are disabled except Reset and L. The system displays the following message: PS-85L communicates with a remote Terminal or PC through the 9 pins RS 232 connector.

**! Serial
Mode!**

Join the Technical Community Today!
<http://www.pantechsolutions.net>

4.10 PROGRAMMING THE 8085 TRAINER KIT:

PROCEDURE 1: TO ENTER THE MNEMONICS

- 1) Initially connect the **9V** adaptor to **J8** connector
- 2) Switch **ON** the **PAN – 85** kit using slide Switch **SW1**
- 3) “**PAN -- 85**” will be displayed on the LCD
- 4) Connect the **Keyboard** in **PS/2** connector
- 5) Depress “**A**” starting address of the program for Ex: **A8500**

For ex: **A8500** enter key

Type the mnemonics **MVI A,12** press Enter key

Type the mnemonics **MVI B, 12** press Enter key and continue the same procedure till the end of the program

ADDRESS	OPCODES	MNEMONICS
8500	3E 12	MVI A,12H
8502	06 12	MVI B,12H
8504	80	ADD B
8505	32 00 86	STA 8600
8508	76	HLT

6) To verify the code depress **D** starting address and depress space bar to see next memory location

For Ex: **D8500** and press spacebar till the end of the program

7) To execute the program Depress “G staring address for Ex: **G8500**.

8) To see the result depress “M result address” for Ex: **M8600**.

9) To view the output in the Register depress 'R' and press enter key in keyboard.

PROCEDURE 2: TO ENTER THE OPCODE

Follow the same procedure till step 4

1) Depress "M" starting address of the program for Ex: **M8500**

For ex: **M8500** press enter

Type the opcode **3E** space bar

Type the opcode **12** space bar and continue the same till the end of the program

ADDRESS	OPCODES	MNEMONICS
8500	3E 12	MVI A,12H
8502	06 12	MVI B,12H
8504	80	ADD B
8505	32 00 86	STA 8600
8508	76	HLT

2) To view the code depress **D** starting address and depress space bar to see next memory location

For Ex: **D8500** and press spacebar till the end of the program

3) To execute the program Depress “G staring address for Ex: **G8500**.

4) To see the result depress “M result address” for Ex: **M8600**.

5) To view the output in the Register depress ‘**R**’ and press enter key in keyboard

Note: 1) “**M**” is used for displaying the result, for Ex: **M8500**

2) “**M**” is used to entering the Opcode.

3) “**M**” is used for entering the data.

Note: There are two ways to enter the program

Join the Technical Community Today!
<http://www.pantechsolutions.net>

- 1) Mnemonics method
- 2) Opcode method

Sample program is given to enter the program in both the methods

CHAPTER – 5 PROGRAMMING DETAILS

PROGRAMMING 8085 OVERVIEW

8085 Microprocessor can access 64kb of external memory using its 16-bit

Address and memory read/write signals.

It has following set of register

A Register (accumulator)

B, C, D, E, H, L

F Flag Register

Of these BC, DE and HL in combination will act as 16 bit registers. Internally it has Arithmetic Logic Unit (ALU),

Join the Technical Community Today!
<http://www.pantechsolutions.net>

Instruction decoder temporary register and timing and control units

INTERRUPTS:

The 8085 have 5 interrupt, which are listed below

External Interrupt : vector to address **9FC1 to 9FC4**

RST 5.5 Interrupt : vector to address **9FC7**

RST 6.5 Interrupt : vector to address **9FC4**

RST 7.5 Interrupt : vector to address **9FC1**

TRAP Interrupt : vector to address **9FCA**

REGISTERS:

- A or Accumulator used for all logical and Arithmetical operations
- B,C,D,E,H and L register are general –purpose register
- Program Counter is 16 bit register capable of addressing from 0000-FFFF

Join the Technical Community Today!
<http://www.pantechsolutions.net>

- Stack Pointer is an 16 bit register it is initialized a value of 9F00
- Status /Flag register keeps track of flags.

BREAK POINT DISPLAY IN LOCAL MODE:

When break point is encountered, all the register values are saved and the Acc. **"A=XX"** Value is displayed in the LCD display. Now use SPACE key to check register values one by one

CHAPTER 6: EXAMPLE PROGRAMS

This chapter describes the technique for development programs in PS-85L board. For the development of programs it is essential to get familiar with the machine code of 8085 Microprocessor or mnemonics of instruction set.

6.1 8 BIT MULTIPLICATION OF TWO DATA'S

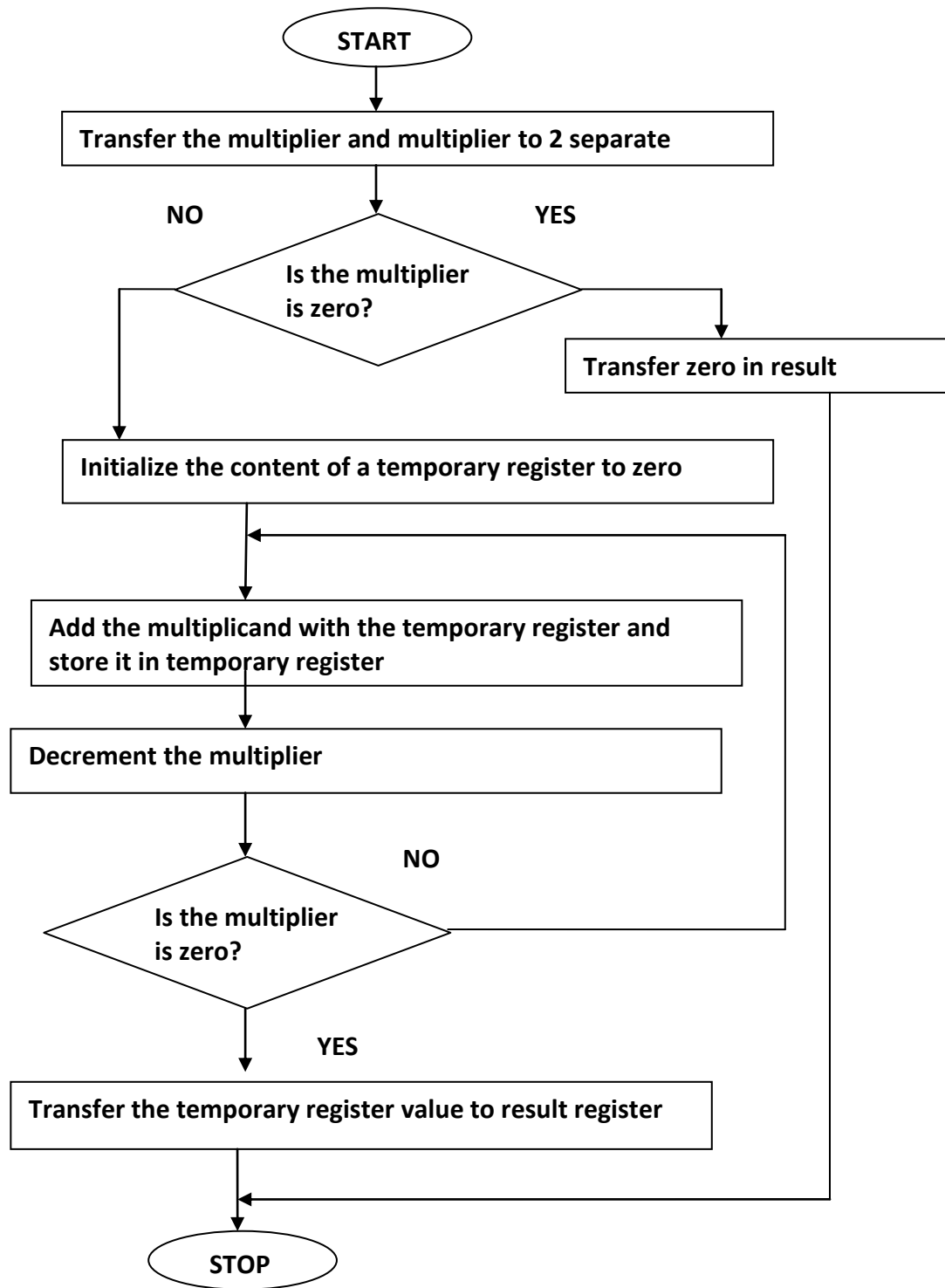
INPUT:

- Multiplicand in 8500.
- Multiplier in 8501.

OUTPUT:

- Result is stored at 8502
- Result is also in Accumulator. If the result exceeds FF value then the carry flag is SET.

FLOW CHART:



PROGRAM :

ADDRES S	OPCOD E	MNEMONIC S	COMMENTS
8100	3A 00 85	LDA 8500	Load 8500 address into Accumulator
8103	47	MOV B,A	Move Accumulator value into B register
8104	3A 01 85	LDA 8501	Load 8501 address into Accumulator
8107	4F	MOV C,A	Move Accumulator value into C register
8108	FE 00	CPI 00	If X2=0 then result is 0 and exit
810A	CA 16 81	JZ 8116	If carry is zero then jump into 8116
810D	AF	XRA A	Clear ACC and flags add the B as many as the C
810E	80	ADD B	Add B register with Accumulator
810F	OD	DCR C	Decrement C register
8110	CA 16 81	JZ 8116	Store the result and exit
8113	C3 0E 81	JMP 810E	Multiply start address = 810E
8116	32 02 85	STA 8502	Store the content of Accumulator into 8502
8119	CF	RST 1	Break point

Join the Technical Community Today!
<http://www.pantechsolutions.net>

6.2 16 BIT MULTIPLICATION OF TWO DATA'S

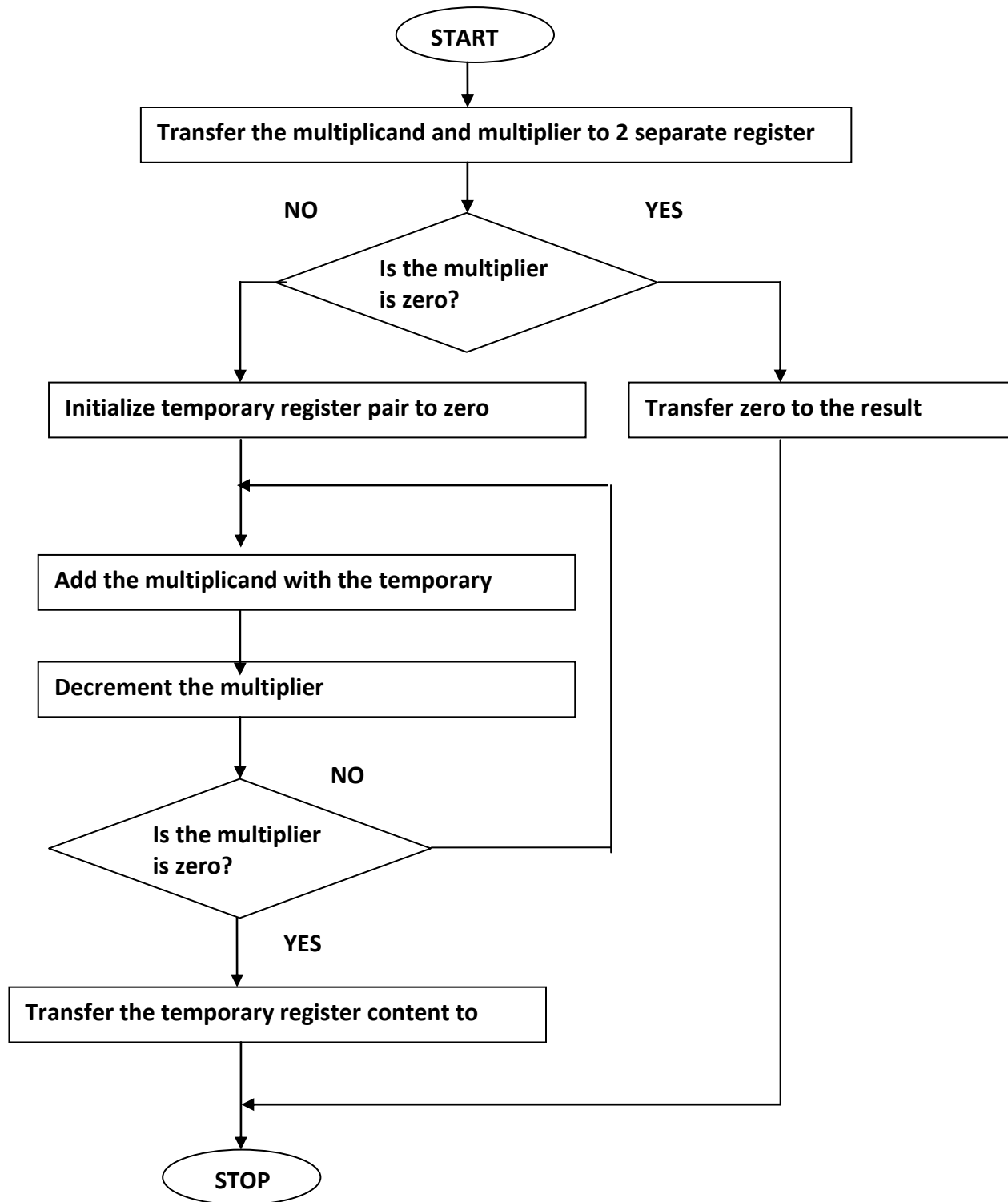
INPUT:

- Multiplicand(X1) Low Byte in 8500, High Byte in 8501.
- Multiplier(X2) in 8502(8 bit value only).

OUTPUT:

- Result is stored at 8503(Low Byte) & 8504(High Byte).
- Result is also in HL. If the result exceeds FF value then the carry flag is SET.

FLOW CHART:



PROGRAM:

ADDRESS	OPCODE	MNEMONICS	COMMENTS
8100	2A 00 85	LHLD 8500	Load 8500 value into HL pair
8103	54	MOV D,H	Move the Highest Byte into D register
8104	5D	MOV E,L	Move the Lowest Byte into E register
8105	3A 02 85	LDA 8502	Load 8502 value into Accumulator
8108	4F	MOV C,A	Move Accumulator value into C register
8109	FE 00	CPI 00	If X2=0 then result is 0 and exit
810B	CA 19 81	JZ 8119	If carry is zero then jump into 8119
810E	21 00 00	LXI H,0000	Initialize the HL register into 0 add HL as many as C
8111	19	DAD D	16 bit addition between HL and DE added value in HL register
8112	0D	DCR C	Decrement the C register
8113	CA 19 81	JZ 8119	Store the result and exit
8116	C3 1181	JMP 8111	Jump multiply(multiply = 8111)
8119	22 03 85	SHLD 8503	Store the content of Accumulator into 8503
811C	CF	RST 1	Break point

6.3 8 BIT DIVISION OF TWO DATA'S

INPUT:

- 8500 – Dividend & 8501 - Divider.

OUTPUT:

- 8502 – Quotient & 8503 - Reminder.

EXAMPLE:

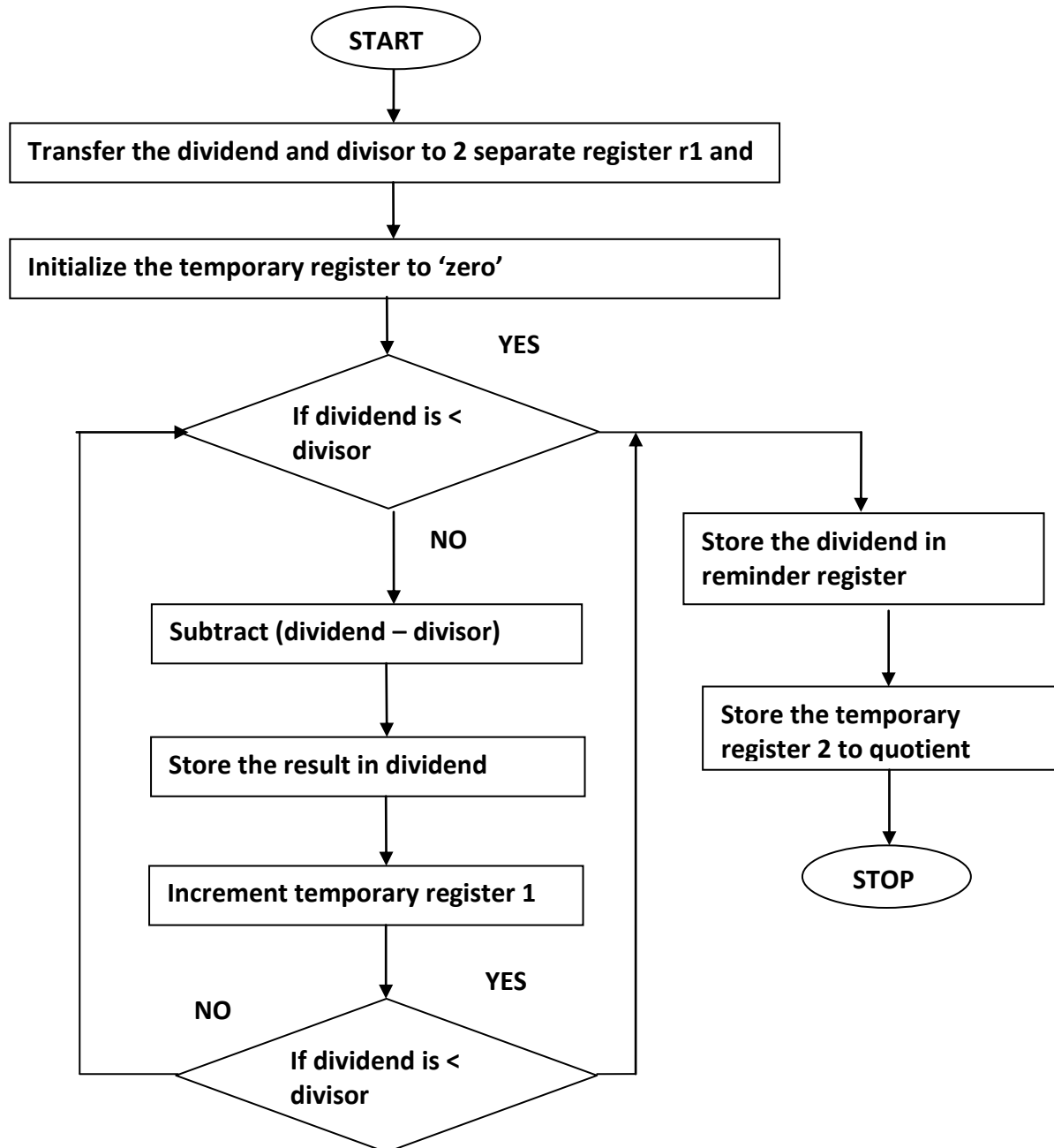
Join the Technical Community Today!
<http://www.pantechsolutions.net>

- 8500 – 0A & 8501 – 02 (that is 0A/02).

RESULT:

- 8502 – 05 (Quotient) & 8503 – 00 (Reminder).

FLOW CHART:



PROGRAM:

ADDRES S	OPCOD E	MNEMONIC S	COMMENTS
8100	3A 01 85	LDA 8501	Load 8501 value into Accumulator
8103	47	MOV B,A	Move Accumulator value into Register B
8104	3A 00 85	LDA 8500	Load 8500 value into Accumulator
8107	0E 00	MVI C,00	Count for quotient
8109	B8	CMP B	Check for A < B
810A	DA 12 81	JC 8112	If A < B then go to store
810D	90	SUB B	Subtract the Register B with Accumulator
810E	0C	INR C	Increment the C register
810F	C3 09 81	JMP 8109	Jump into 8109
8112	32 03 85	STA 8503	Store Reminder value into 8503
8115	79	MOV A,C	Move Register C value into Accumulator
8116	32 02 85	STA 8502	Store the Quotient value into 8502
8119	CF	RST 1	Break point

Join the Technical Community Today!
<http://www.pantechsolutions.net>

6.4 16 BIT DIVISION OF TWO DATA'S

INPUT:

- 9200 – Dividend (Low Byte) & 9201 – Dividend (High Byte).
- 9202 – Divider (Low Byte)

OUTPUT:

- 9300 – Quotient (Low Byte) & 9301 – Quotient (High Byte).
- 9302 – Reminder (Low Byte) & 9303 – Reminder (High Byte).

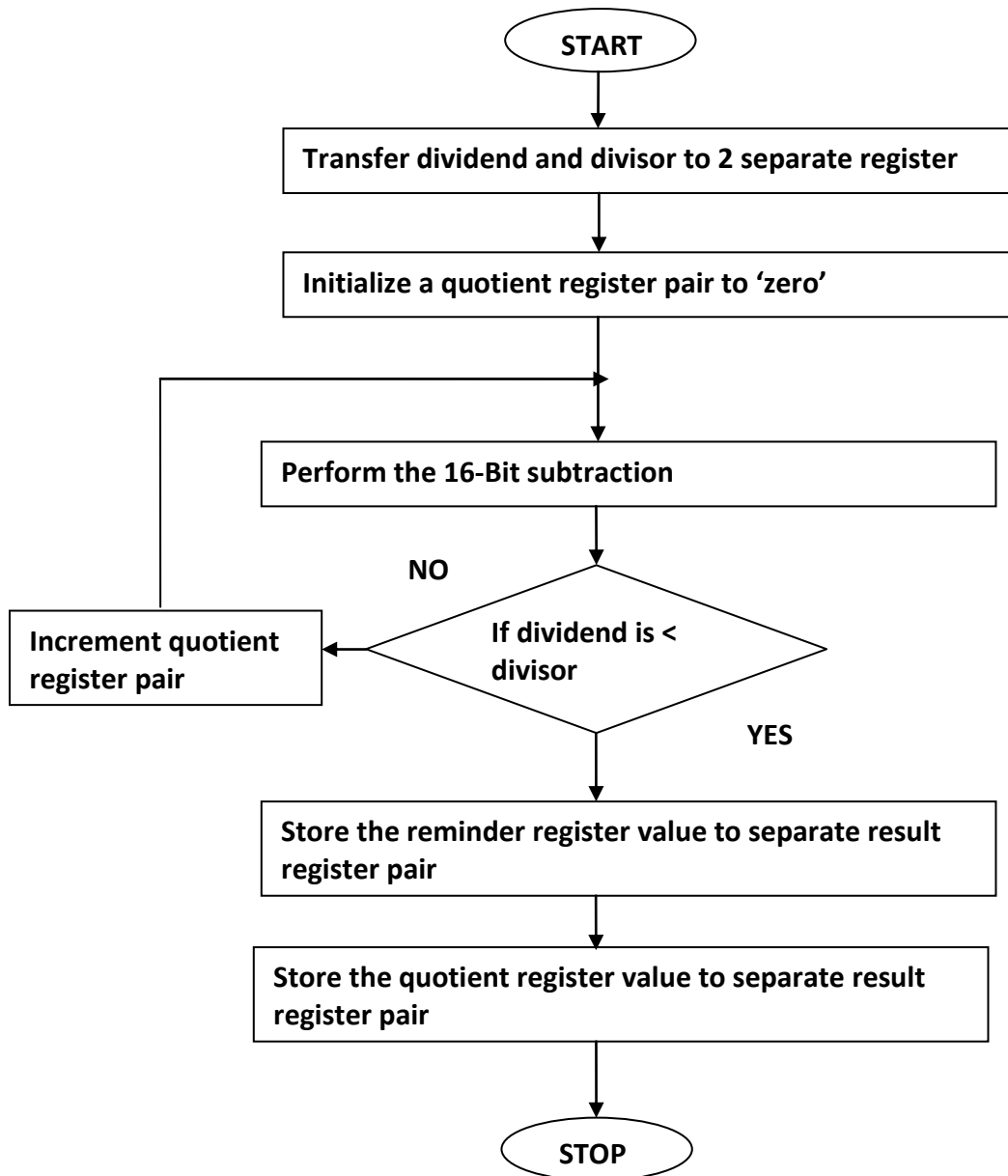
EXAMPLE:

- 9200 – 0A (Low Byte) & 9201 – 0A (High Byte).
- 9202 – 02 (Low Byte)

RESULT:

- 9300 – 05 (Low Byte) & 9301 – 05 (High Byte) → Quotient.
- 9302 – 00 (Low Byte) & 9303 – 00 (High Byte) → Reminder.

FLOW CHART



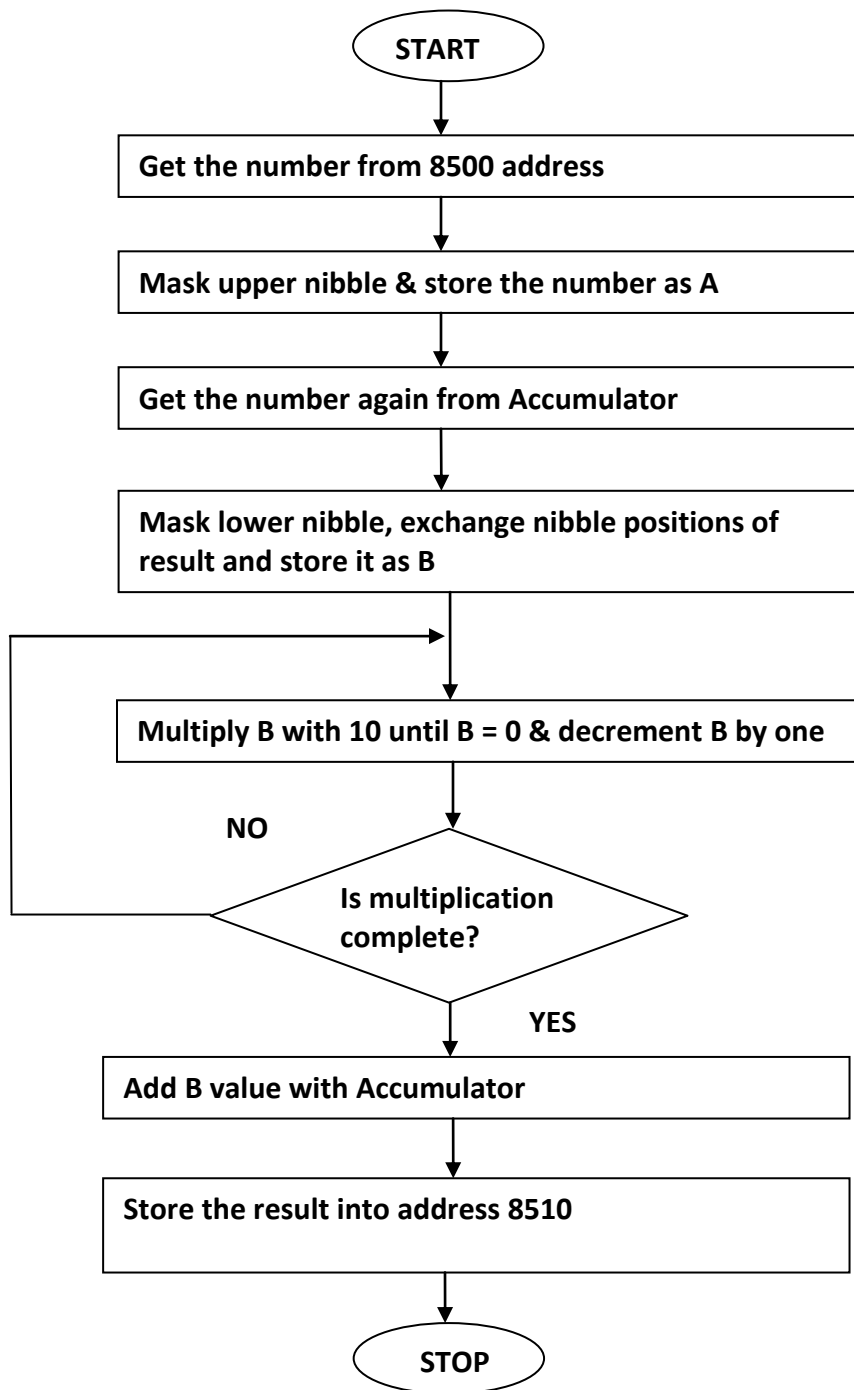
PROGRAM:

ADDRESS	OPCODE	MNEMONICS
9100	2A 00 92	LHLD 9200
9103	3A 02 92	LDA 9202
9106	4F	MOV C,A
9107	11 00 00	LXI D,0000
910A	7D	BACK : MOV A,L
910B	91	SUB C
910C	6F	MOV L,A
910D	D2 11 91	JNC SKIP
9110	25	DCR H
9111	13	SKIP : INX D
9112	7C	MOV A,H
9113	FE 00	CPI 00
9115	C2 0A 91	JNZ BACK
9118	7D	MOV A,L
9119	B9	CMP C
911A	D2 0A 91	JNC BACK
911D	22 02 93	SHLD 9302
9120	EB	XCHG
9121	22 00 93	SHLD 9300
9124	76	HLT

6.5 2 DIGIT BCD TO BINARY (HEXA decimal) CONVERSATION

Join the Technical Community Today!
<http://www.pantechsolutions.net>

FLOW CHART:



INPUT: 8500 – BCD value.

OUTPUT: Result is stored at 8510.

PROGRAM

ADDRES S	OPCODE	MNEMONICS	COMMENTS
8100	3A 00 85	LDA 8500	Load 8500 value into Accumulator
8103	47	MOV B,A	Move Accumulator value into B register
8104	E6 0F	ANI 0FH	Mask upper nibbles & store it into Accumulator
8106	4F	MOV C,A	Move Accumulator value into C register
8107	78	MOV A,B	Load Register B value into Accumulator
8108	E6 F0	ANI F0H	Mask lower nibbles & store it into Accumulator
810A	0F	RRC	Convert MSB into unpacked B register
810B	0F	RRC	
810C	0F	RRC	
810D	0F	RRC	
810E	47	MOV B,A	Move Accumulator value into B register
810F	AF	XRA A	The content of ACC X-OR with the content of register
8110	16 0A	MVI D,0AH	Store multiplier 10 value into D register
8112	82	ADD D	Add the 10 value with Accumulator
8113	05	DCR B	Decrement the B register by one
8114	C2 12 81	JNZ 8108	If multiplication is not complete jump into 8108
8117	81	ADD C	Else Add with C register
8118	32 10 85	STA 8510	Store Accumulator value into 8510
811B	CF	RST 1	Break point

6.6 BINARY (Hex) TO ASCII CONVERSION

INPUT:

- Binary Hex Byte in 8500.

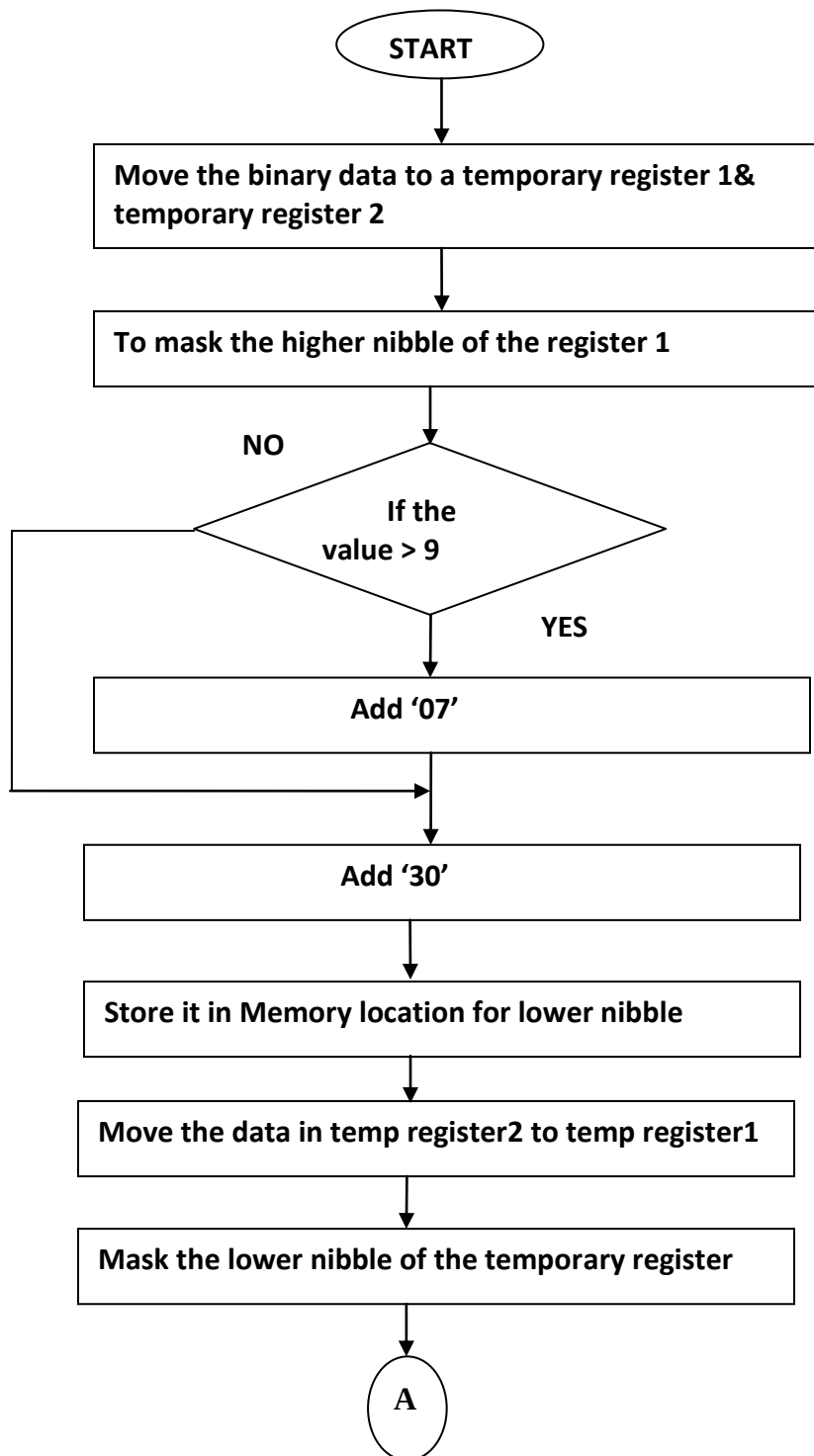
OUTPUT:

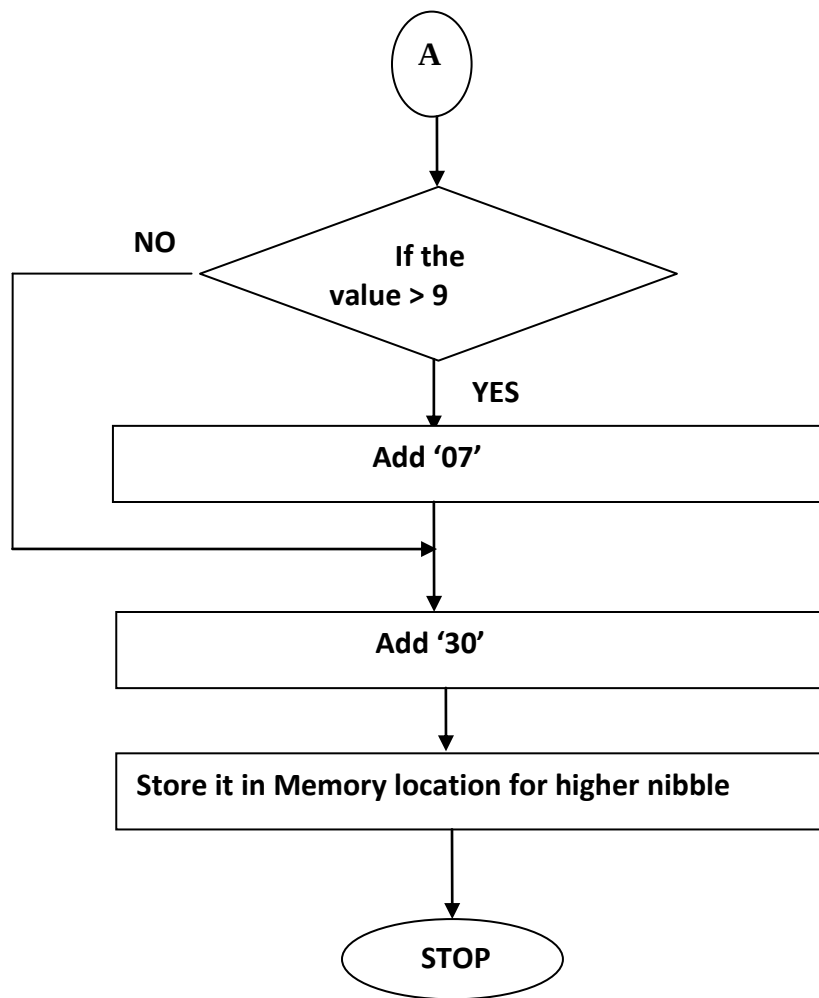
- ASCII of higher nibble in 8501 & lower nibble in 8502.

EXAMPLE:

- Input, 8500 – A1.
- Output, 8501 – 41 ASCII of 'A' & 8502 – 31 ASCII of '1'.

FLOW CHART:





PROGRAM

ADDRESS	OPCODE	MNEMONICS	COMMENTS
8100	21 01 85	LXI H,8500	Load 8500 value into H register
8103	7E	MOV A,M	Move M register value into Accumulator
8104	47	MOV B,A	Move Accumulator value into B register
8105	23	INX H	Increment the H register by one
8106	0F	RRC	Each binary bit of the ACC is rotated right by one
8107	0F	RRC	
8108	0F	RRC	
8109	0F	RRC	
810A	CD 15 81	CALL 8115	Now convert the lower nibble to ASCII
810D	77	MOV M,A	Store that ASCII at 8501
810E	23	INX H	Now HL=8502, Increment H register by one
810F	78	MOV A,B	Move Register B value into Accumulator
8110	CD 15 81	CALL 8115	Now convert the lower nibble to ASCII
8113	77	MOV M,A	Store that ASCII at 8502
8114	CF	RST 1	Break point

This function converts the lower nibble of Acc to its ASCII value

Join the Technical Community Today!
<http://www.pantechsolutions.net>

ADDRESS	OPCODE	MNEMONICS	COMMENTS
8115	E6 0F	ANI 0F	Mask the higher nibble
8117	FE 0A	CPI 0A	Check is it greater then 9
8119	DA 1E 81	JC 811E	If A not > 9 then simply add 30
811C	C6 07	ADI 07	If A not > 9 then add 37 to get the ASCII value
811E	C6 30	ADI 30	
8120	C9	RET	Return into Main

6.7 ASCENDING ORDER

INPUT:

- 8500 -No of elements in an array & 8501- starting address of this array.

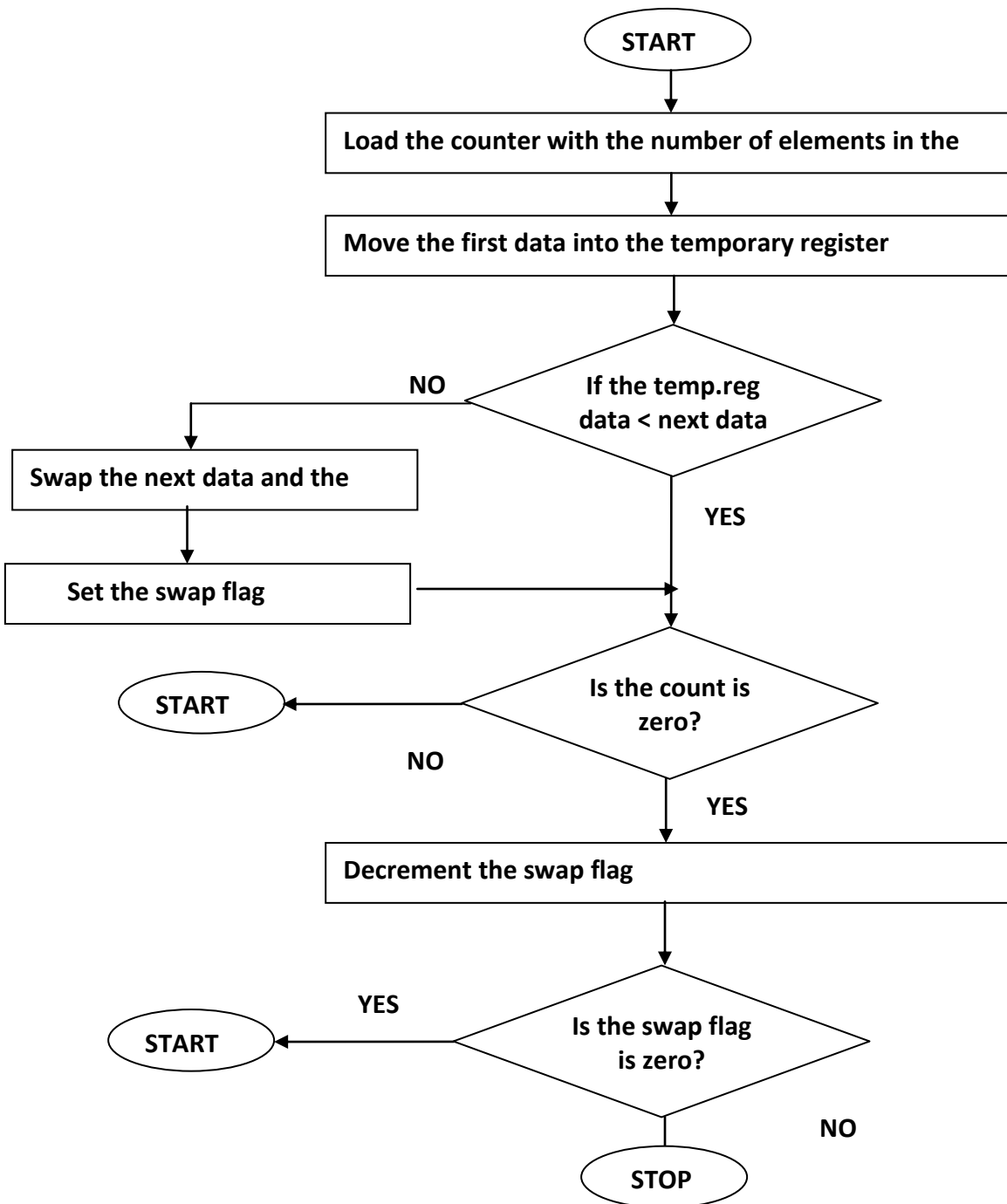
OUTPUT:

- The array elements are stored in ascending order & store in same array.

EXAMPLE:

- Input, 8500 – 05(count). 8501 - 08 04 03 07 01.
- Output, 8501 – 01 03 04 07 08.

FLOW CHART:



PROGRAM:

Join the Technical Community Today!
<http://www.pantechsolutions.net>

ADDRESS	OPCODE	MNEMONICS	COMMENTS
8100	06 00	MVI B,00	Initialize 00 into B register
8102	21 00 85	LXI H,8500	Load 8500 value into H register
8105	4E	MOV C,M	Load the count value into the C register
8106	0D	DCR C	If C=1 no process
8107	CA 20 81	JZ 8120	Jump into 8120(program end)
810A	23	INX H	Increment the H register by one
810B	7E	MOV A,M	Get the byte pointed by m pointer
810C	23	INX H	Move the m pointer to next byte
810D	BE	CMP M	Compare with ACC
810E	DA 18 81	JC 8118	If Acc < m then no swapping
8111	56	MOV D,M	Else swap the bytes
8112	77	MOV M,A	Load Accumulator value into M
8113	2B	DCX H	Decrement H value by one
8114	72	MOV M,D	Load D register value into pointer
8115	23	INX H	Now move m pointer to the next byte
8116	06 01	MVI B,01	Set the swap flag
8117	0D	DCR C	Decrement the count
8119	C2 0B 81	JNZ 810B	If C is not equal 0 then jump 810B
811C	5	DCR B	After C=0 check the swap flag.
811D	CA 00 81	JZ 8100	If it is set then once again continue the process at the final stage B will not set so the pgm will terminate.
8120	CF	RST1	Break point.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

6.8 DESCENDING ORDER

INPUT:

- 8500 -No of elements in an array & 8501- starting address of this array.

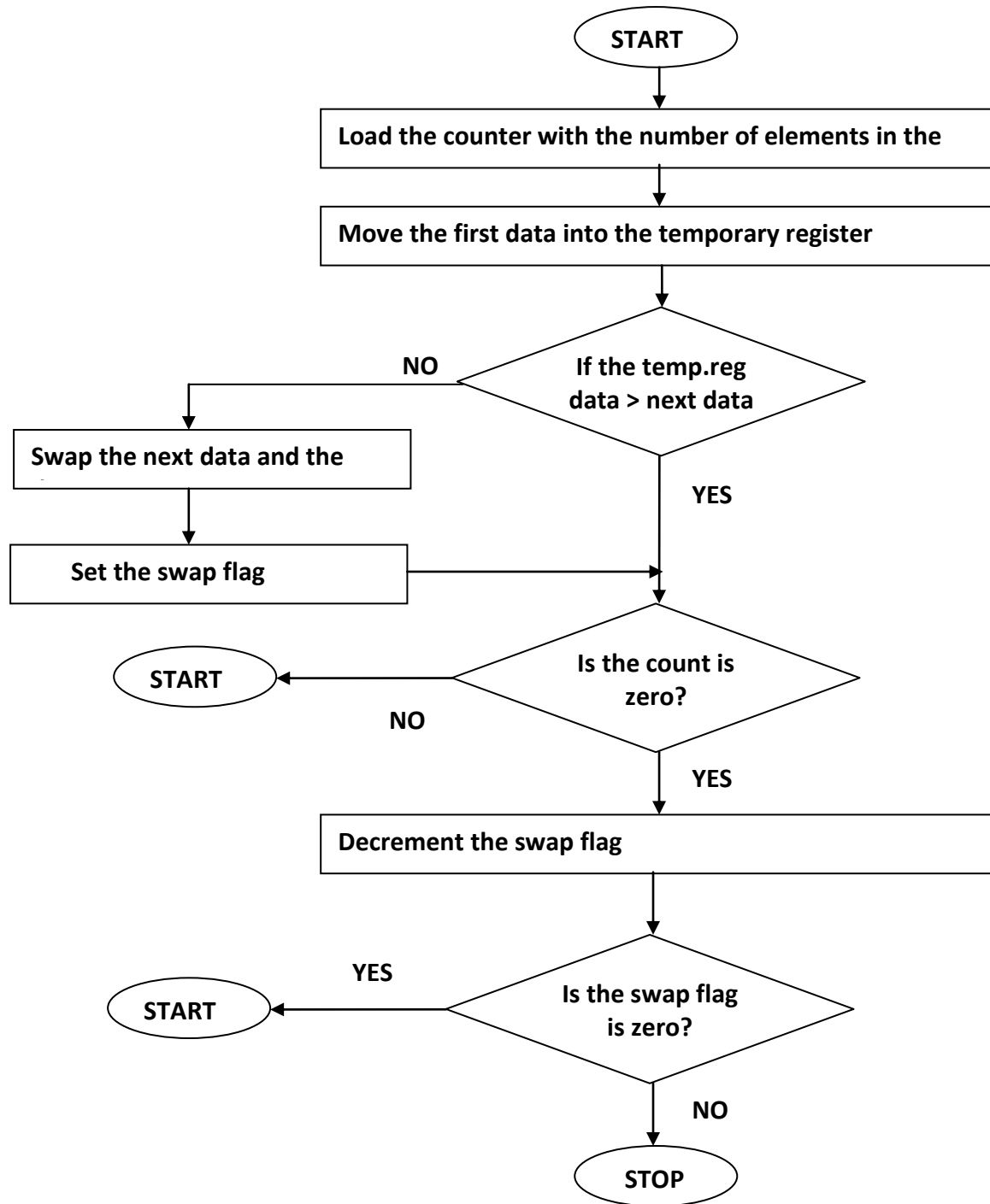
OUTPUT:

- The array elements are stored in descending order & store in same array.

EXAMPLE:

- Input, 8500 – 05(count). 8501 - 08 04 03 07 01.
- Output, 8501 – 08 07 04 03 01.

FLOW CHART:



PROGRAM:

ADDRESS	OPCODE	MNEMONICS	COMMENTS
8100	06 00	MVI B,00	Initialize 00 into B register
8102	21 00 85	LXI H,8500	Load 8500 value into H register
8105	4E	MOV C,M	Load the count value into the C register
8106	0D	DCR C	If C=1 no process
8107	CA 20 81	JZ 8120	Jump into 8120(program end)
810A	23	INX H	Increment the H register by one
810B	7E	MOV A,M	Get the byte pointed by m pointer
810C	23	INX H	Move the m pointer to next byte
810D	BE	CMP M	Compare with ACC
810E	DA 18 81	JNC 8118	If Acc > m then no swapping
8111	56	MOV D,M	Else swap the bytes
8112	77	MOV M,A	Load Accumulator value into M
8113	2B	DCX H	Decrement H value by one
8114	72	MOV M,D	Load D register value into pointer
8115	23	INX H	Now move m pointer to the next byte
8116	06 01	MVI B,01	Set the swap flag
8117	0D	DCR C	Decrement the count
8119	C2 0B 81	JNZ 810B	If C is not equal 0 then jump 810B
811C	5	DCR B	After C=0 check the swap flag.
811D	CA 00 81	JZ 8100	If it is set then once again continue the process at the final stage B will not set so the pgm will terminate.
8120	CF	RST1	Break point.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

6.9 FIND THE BIGGEST NUMBER IN AN ARRAY

INPUT:

- 8500 -No of elements in an array & 8510- starting address of this array.

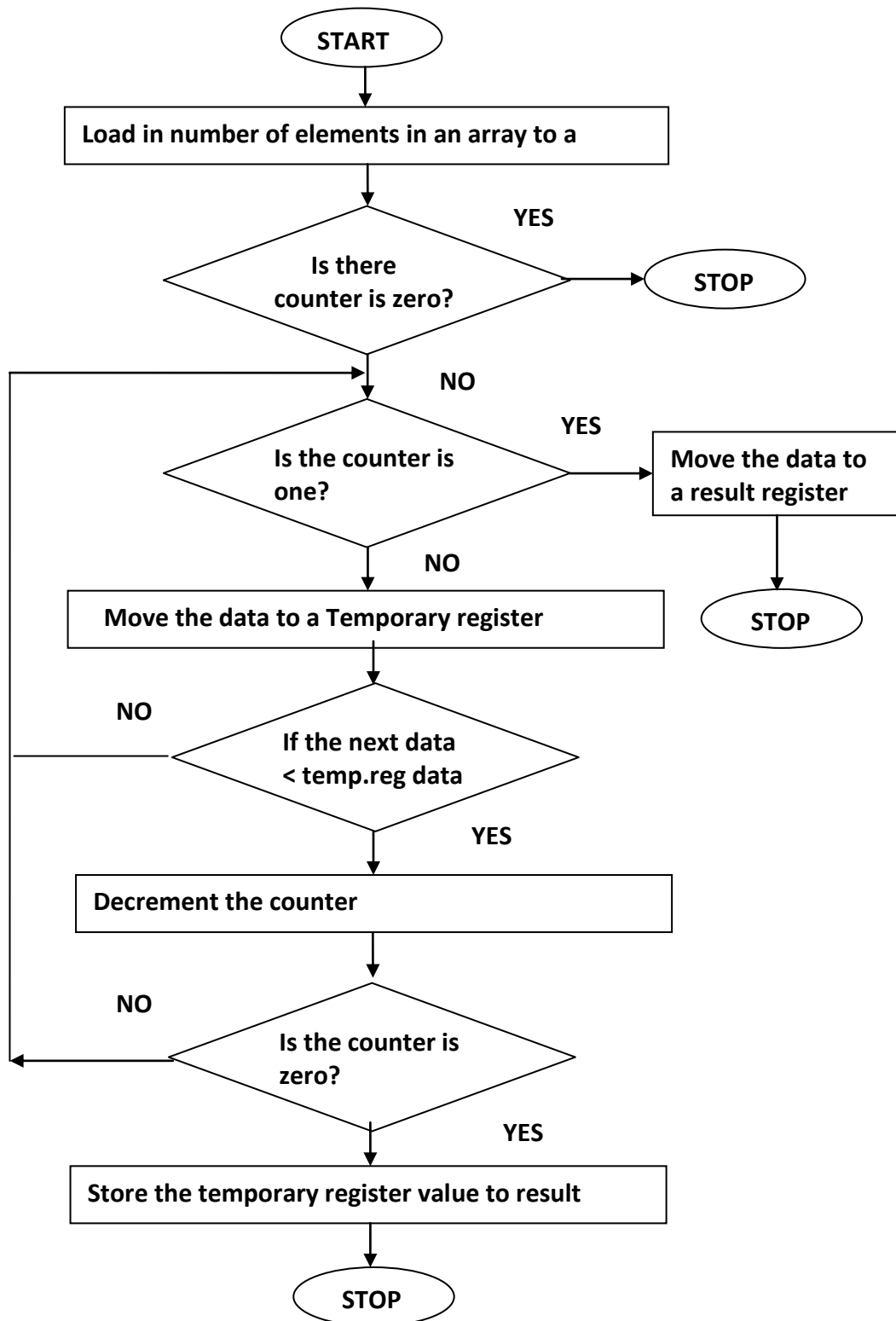
OUTPUT:

- The Biggest number stored ACC and 8501.

EXAMPLE:

- Input, 8500 – 05(count). 8510 - 08 04 03 07 01.
- Output, 8501 – 08.

FLOW CHART:



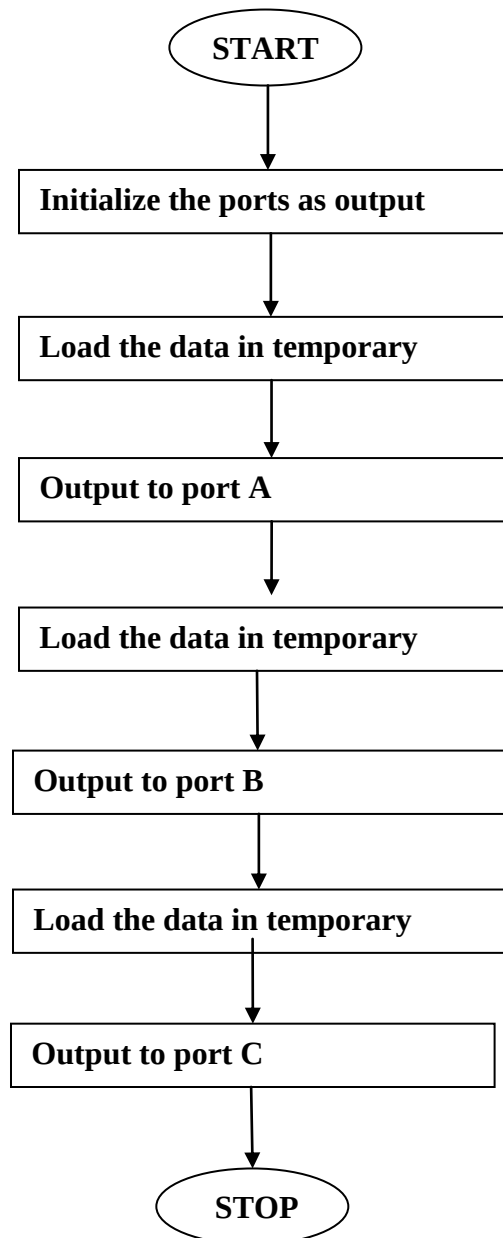
PROGRAM:

ADDRESS	OPCODE	MNEMONICS	COMMENTS
8100	21 10 85	LXI H,8510	Load 8510 value into H register
8103	3A 00 85	LDA 8500	Load the count value into 8500 address
8106	4F	MOV C,A	Load the count value into C Register
8107	0D	DCR C	Decrement the count value
8108	DA 20 81	JC 8120	If the count = 1 then continue
810B	CA 1C 81	JZ 811C	Get the num and exit
810E	7E	MOV A,M	Get the byte pointed by M pointer
810F	23	INX H	Move the pointer to the next byte
8110	BE	CMP M	Compare with Accumulator
8111	D2 15 81	JNC 8115	If ACC > M then no swapping
8114	7E	MOV A,M	Else get the large no in to Accumulator
8115	0D	DCR C	Decrement the count value
8116	C2 0F 81	JNZ 810F	If C ≠ 0 then jump 81F0
8119	C3 0F 81	JMP 811D	Else Jump into 811D
811C	7E	MOV A,M	Store the biggest number into Accumulator
811D	32 01 85	STA 8501	The result is stored into 8501
8120	CF	RST 1	Break Point

6.10 VERIFYING THE PORTS OF 8255

Join the Technical Community Today!
<http://www.pantechsolutions.net>

FLOW CHART:



PROGRAM:

ADDRESS	OPCODE	MNEMONICS	COMMENTS
8100	3E 80	MVI A,80	Load the value 80 into Accumulator
8102	D3 23	OUT 23	Output to control register
8104	3E 44	MVI A,44	Load data 44 into Accumulator
8106	D3 20	OUT 20	Output to port A
8108	3E 88	MVI A,88	Load data 88 into Accumulator
810A	D3 21	OUT 21	Output to port B
810C	3E FF	MVI A,FF	Load data FF into Accumulator
810E	D3 22	OUT 22	Output to port C
8110	CF	RST 1	Break point

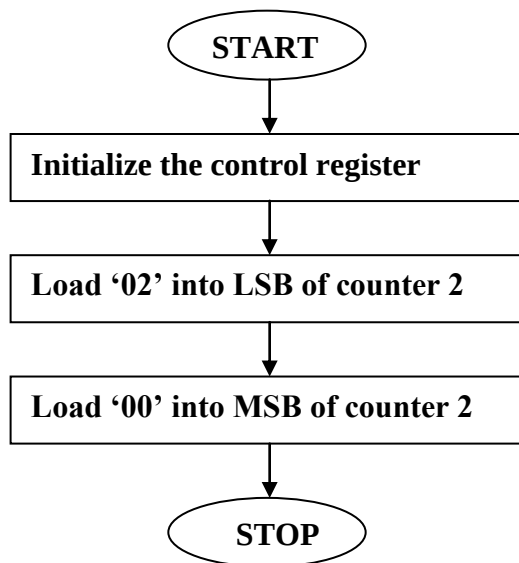
6.11 SQUARE WAVE GENERATION USING 8253/54

OUTPUT:

- This program will generate the square wave of Frequency 625 KHz.
- Observe the wave form at pin no 10 of 8253 IC.

FLOW CHART:

Join the Technical Community Today!
<http://www.pantechsolutions.net>



PROGRAM:

ADDRES S	OPCODE E	MNEMONIC S	COMMENTS
8100	3E B7	MVI A,B7	Load the control word into Accumulator
8102	D3 13	OUT 13	O/P the ACC data into control register (I/O addr13)
8104	3E 02	MVI A,02	Load the LSB count into the Accumulator
8106	D3 12	OUT 12	O/P the ACC data into counter2
8108	3E 00	MVI A,00	Load MSB count into Accumulator
810A	D3 12	OUT 12	O/P the ACC data into counter2
810C	CF	RST1	Break point

ADDITIONAL PROGRAMS ON 8085 TRAINER KIT

1) 8-BIT ADDITION

OBJECTIVE

Join the Technical Community Today!
<http://www.pantechsolutions.net>

To add two 8 - bit numbers in memory and store the result in memory.

EXAMPLE

The two data to be added are at 8650 and 8651. The result is stored at 8652.

DATA : (8650) = 23

(8651) = 35

RESULT : (8652) = 58.

PROGRAM

ADDRESS	OPCODE	MNEMONICS
8500	3A 50 86	LDA 8650
8503	47	MOV B,A
8504	3A 51 86	LDA 8651
8507	80	ADD B
8508	32 52 86	STA 8652
850B	76	HLT

PROCEDURE

1. Key in the opcodes from the address specified.

2. Enter data at 8650 and 8651 as specified in the Example.
3. Execute the program and check for the result at 8652.
4. Change data at 8650 and 8651 and execute each time and check for result.

2) 8-BIT SUBTRACTION

OBJECTIVE

To subtract two 8-bit numbers in memory and to store the result also in memory.

EXAMPLE

Let the data to be subtracted be 24 from 49.

DATA: (8650) = 49

(8651) = 24

RESULT: (8652) = 25

PROGRAM

ADDRESS	OPCODE	MNEMONICS
8500	21 50 86	LDA 8650
8503	7E	MOV A,M
8504	23	INX H
8505	96	SUB M
8506	23	INX H
8507	77	MOV M,A
8508	76	HLT

PROCEDURE

1. Key in the opcodes from the address specified.
2. Enter data that is needed for execution at 8650 and 8651.
3. Execute the program and check for results at 8652.
4. Try changing data and check for results each time.

3) ONE'S COMPLEMENT

OBJECTIVE

To find the One's Complement of the data at memory location 8650 and store the result at 8651.

EXAMPLE

The example given is to find the One's complement of 8F and store it in memory location 8651.

The number 8F is at memory location 8650.

DATA : (8650) = 1000 1111 B

RESULT : (8651) = 0111 0000 B

PROGRAM

ADDRESS	OPCODE	MNEMONICS
8500	3A 50 86	LDA 8650
8503	2F	CMA
8504	32 51 86	STA 8650
8507	76	HLT

PROCEDURE

1. Key in the opcodes into memory from 8600.
2. Enter data to be complemented at 8650.
3. Execute the program and check for results.

4. Try changing data at 8650 and execute the program each time, and check results.

4) MASK OFF MOST SIGNIFICANT FOUR BITS

OBJECTIVE

To place the LN of the contents of memory location 8650 in memory location 8651. Clear the HN of the number.

EXAMPLE

The contents of 8650 are B8. The result will be stored at 8651.

DATA: (8650) = B8

RESULT: (8651) = 08

PROGRAM :

ADDRESS	OPCODE	MNEMONICS
8500	3A 50 86	LDA 8650
8503	E6	ANI 0F
8504	32 51 86	STA 8650
8507	76	HLT

Join the Technical Community Today!
<http://www.pantechsolutions.net>

PROCEDURE:

1. Key in the opcodes from the address specified.
2. Enter data at 8650.
3. Execute the program and check for result at 8651.
4. Change data at 8650 and execute each time and check for result at 8651.

5) LOGICAL OPERATIONS

OBJECTIVE

To learn about logical instructions available with 8085A like AND, OR, XOR, NOT.

AND	OR	XOR	NOT
X Y Z	X Y Z	X Y Z	X Z
0 0 0	0 0 0	0 0 0	0 1
0 1 0	1 0 1	0 1 1	1 0
1 0 0	0 1 1	1 0 1	
1 1 1	1 1 1	1 1 0	

Where X, Y are two inputs, Z is the output.

EXAMPLE:

The two numbers to be XORed are at memory locations 8650 and 8651 and the XORed result will be stored at 8652.

DATA: (8650) = 43 (8651) = 64

RESULT: 0100 0011B

0110 0100B (XOR)

0010 0111B

(8652) = 27.

PROGRAM:

ADDRESS	OPCODE	MNEMONICS
8500	3A 50 86	LDA 8650
8503	47	MOV B,A
8504	3A 51 86	LDA 8651
8507	A8	XRA B
8508	32 52 86	STA 8652
850B	76	HLT

PROCEDURE:

1. Key in the opcodes from the address specified.
2. Enter data to be XORed at 8650 and 8651.
3. Execute the program and check for results at 8652.

6) ASCII TO DECIMAL CONVERSION

OBJECTIVE

To convert the ASCII number in memory to its equivalent decimal number.

EXAMPLE

Let the ASCII number be at location 8650 and the result be stored at 8651.

DATA: (8650) = 35

RESULT: (8651) = 05.

DATA: (8650) = 3A

RESULT: (8651) = FF.

So if the decimal number got is not a valid one, FF will be stored as the result to indicate the error.

PROGRAM:

ADDRESS	OPCODE	MNEMONICS
8500	21 50 86	LXI H, 8650
8503	7E	MOV A, M
8504	DE 30	SUI 30
8506	FE 0A	CPI 0A
8508	DA 0D 85	JC LOOP
850A	3E FF	MVI A,FF
850C	23	LOOP : INX H
850D	77	MOV M, A
850F	76	HLT

PROCEDURE:

1. Key in the opcodes from the address specified.
2. Enter data at 8650.
3. Execute the program and check at 8651 for result.
4. Try changing data and check result each time at 8651.

7) BCD TO HEX CONVERSION

OBJECTIVE:

Join the Technical Community Today!
<http://www.pantechsolutions.net>

Convert two BCD numbers in memory to the equivalent hex number.

EXAMPLE:

The BCD digits are at locations 8650 and 8651 and result will be stored at 8652.

DATA: (8650) = 02 (MSD)

(8651) = 09 (LSD)

RESULT: (8652) = 1D H = 29 D

PROGRAM

ADDRESS	OPCODE	MNEMONICS
8500	21 50 86	LXI H, 8650
8503	7E	MOV A
8504	87	ADD A
8505	47	MOV B,A
8506	87	ADD A
8507	87	ADD A
8508	80	ADD B
8509	23	INX H
850A	86	ADD M
850B	23	INX H
850C	77	MOV M, A
850D	76	HLT

Join the Technical Community Today!
<http://www.pantechsolutions.net>

PROCEDURE

1. Key in the opcodes from the address specified.
2. Enter data at 8650 and 8651 for execution.
3. Execute the program and check for result at 8652.
4. Try changing data and check result each time.

8) HEX TO BINARY FORM

OBJECTIVE

Convert an 8-bit hex number to its binary form and store in memory.

EXAMPLE

Let the data be 5A.

DATA: 5A = 0101 1010 B

RESULT: (8650) = 0 (LSB) (8651) = 1

(8652) = 0 (8653) = 1

(8654) = 1 (8655) = 0

(8656) = 1 (8657) = 0 (MSB)

PROGRAM:

Join the Technical Community Today!
<http://www.pantechsolutions.net>

ADDRESS	OPCODE	MNEMONICS
8500	21 50 86	LXI H,8650
8503	06 08	MVI B,08
8505	3E 5A	MVI A,5A
8507	0F	LOOP : RRC
8508	DA 10 86	JC LOOP1
850B	36 00	MVI M,00
850D	C3 12 85	JMP COMMON
8510	36 01	LOOP1: MVI M,01
8512	23	COMMON: INX H
8513	05	DCR B
8514	C2 07	JNZ LOOP
8517	76	HLT

PROCEDURE

1. Key in the opcodes from the address specified.
2. Execute the program and check for results at locations 8650 through 8657.

Did you enjoy the read?

Pantech solutions creates information packed technical documents like this one every month. And our website is a rich and trusted resource used by a vibrant online community of more than 1,00,000 members from organization of all shapes and sizes.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

What do we sell?

Our products range from Various Microcontroller development boards, DSP Boards, FPGA/CPLD boards, Communication Kits, Power electronics, Basic electronics, Robotics, Sensors, Electronic components and much more . Our goal is to make finding the parts and information you need easier and affordable so you can create awesome projects and training from Basic to Cutting edge technology.

Join the Technical Community Today!
<http://www.pantechsolutions.net>